

considered this to be a viable grounds for rejection, the present final Office Action should be withdrawn and another Office Action issued that explicitly rejected the claims under 35 U.S.C. § 101. The Examiner agreed to consult his supervisor about this matter. Since a subsequent Interview Summary mailed March 2, 2006, did not discuss a § 101 rejection, applicants assume that this grounds for rejection has been withdrawn.

The following discussions of the applicants' invention and the cited and applied references are not provided to define the scope or interpretation of any of the claims of this application. Instead, these discussions are provided to help the United States Patent and Trademark Office better appreciate the important claim distinctions discussed herein.

Rejection of Claims 1, 3, 5-8, 10, 11, 14-16, 18-28, and 30-33 Under 35 U.S.C. § 102(e)

As noted above, Claims 1, 3, 5-8, 10, 11, 14-16, 18-28, and 30-33 were rejected under 35 U.S.C. § 102(e) by the Office Action for "being anticipated by Kossatchev." Each independent claim and the claims dependent therefrom are addressed separately as follows.

Claims 23-28 and 30-33

Independent Claim 23 recites, in its entirety:

23. A computer-readable storage medium having stored thereon a data structure, the data structure comprising:

a first data field containing data representing an application table, the application table comprising an application table entry; and

a second data field containing data representing a parameter table, the parameter table comprising a parameter table entry.  
(Emphasis added.)

Kossatchev does not teach or suggest a data structure comprising a first data field containing data representing an application table and a second data field containing data representing a parameter table. Kossatchev purportedly discloses a means 12 for generating specifications of the procedure interface 4 and a test source generator 14 that generates test

LAW OFFICES OF  
CHRISTENSEN O'CONNOR JOHNSON KINDNESS<sup>PLLC</sup>  
1420 Fifth Avenue  
Suite 2800  
Seattle, Washington 98101  
206.682.8100

source code based on the specifications. The specifications and test sources are stored in a repository 16 (Col. 3, lines 6-13). The repository 16 is not even used to store a table, but the specifications and test sources, let alone to store a first and a second data field containing data representing an application table and a parameter table, respectively. With reference to Figure 1 of Kossatchev, test sources are used to generate the test suite 22. The test suite 22 is a set of programs and test data intended for use in verifying the target procedure interface 4 (Col. 3, lines 14-16). Test case parameter generator 32 generates test case parameter sources for generating test case parameters. That is, the test case parameter generator 32 generates constant arrays and programs that generate and select needed test case parameters (Col. 4, lines 11-15). As illustrated in Figure 1, test suite 22 and test case parameters are not contained in one data structure, as recited by Claim 13. Kossatchev does not teach or suggest the use of an application table. Kossatchev discloses test suite 22, which is a set of programs and test data (Col. 3, line 15). As is well known in the art, a table is a data structure comprising multiple entries that can be accessed using an index. Therefore, an application table, as recited by Claim 23, is different from a set of programs and test data, as disclosed by Kossatchev. Therefore, it is respectfully submitted that Claim 23 is allowable.

Claim 24 depends from Claim 23 and is submitted to be allowable for at least the same reasons as presented for Claim 23. Additionally, Claim 24 recites "a third data field containing data representing a global initialize function; a fourth data field containing data representing a global terminate function." Kossatchev does not teach or suggest a third data field containing data representing a global initialize function. Furthermore, Kossatchev does not teach or suggest that the functions provided to initialize the procedure interface 4 are global. Those skilled in the art appreciate that the term "global" implies general applicability of a procedure to multiple similar software entities. Similarly, Kossatchev does not teach or suggest the use of a global

terminate function. Therefore, Claim 24 is further submitted to be allowable for the additional reasons discussed above.

Claim 25 depends from Claim 24 and is submitted to be allowable for at least the same reasons discussed above with respect to Claim 24. Additionally, Claim 25 recites "a sixth data field containing data representing an application test function." (Emphasis added.) Kossatchev does not teach or suggest an application test function represented by data in a data field. Kossatchev discloses specifying the "behavior of a group of parallel procedures to be tested in a parallel mode separately from consecutive procedures." Kossatchev, Col. 1, lines 44-46. Therefore, Claim 25 is further submitted to be allowable for the reasons additional discussed above.

Claim 26 depends from Claim 24 and is submitted to be allowable for at least the same reasons discussed above with respect to Claim 24.

Claim 27 depends from Claim 26 and is submitted to be allowable for at least the same reasons discussed above with respect to Claim 26. Additionally, Claim 27 recites "a seventh data field containing data representing an application post function." (Emphasis added.) Kossatchev does not teach or suggest an application post function represented by data in a data field. Kossatchev discloses a test suite 22 that executes tests on a system under test 3 and analyzes results of the tests to verify the procedure interface 4. Kossatchev, Col. 3, lines 60-63. Therefore, Claim 27 is further submitted to be allowable for the additional reasons discussed above.

Claim 28 depends from Claim 27 and is allowable for at least the same reasons discussed above with respect to Claim 27. Additionally, Claim 28 recites "an eighth data field containing data representing an application post test function." (Emphasis added.) Kossatchev does not teach or suggest an application post test function represented by data in a data field. Kossatchev

discloses a test driver that checks the "correctness of the target procedure execution results." Kossatchev, Col. 4, lines 55-60. Therefore, Claim 28 is further submitted to be allowable for the additional reasons discussed above.

Claim 30 depends from Claim 23 and is submitted to be allowable for at least the same reasons discussed above with respect to Claim 23. Additionally, Claim 30 recites "a third data field containing data representing a name of a parameter; a fourth data field containing data representing a type of the parameter; and a fifth data field containing data representing a value of the parameter." (Emphasis added.) Kossatchev does not teach or suggest a name of the parameter, a type of the parameter, and a value of the parameter represented by data in a data field. Kossatchev discloses that "the test case parameter generator 32 generates constant arrays and programs that generate and select needed test case parameters." (Emphasis added.) Kossatchev, Col. 4, lines 13-16. Kossatchev further discloses that "[a] test case is an instance of a tested procedure. A test case is defined by a procedure name and its parameters. ...The test drivers use the test case parameters and execute test cases on the SUT 3 to verify the procedure interface 4." (Emphasis added.) Kossatchev, Col. 4, lines 25-29. Kossatchev discloses the use of parameters as part of the definition of a test case, in contrast to Claim 30, which recites name, type, and value of a parameter represented by data in a data field. Therefore, Claim 30 is further submitted to be allowable for the additional reasons discussed above.

Claims 31, 32, and 33 depend from Claim 23 and are submitted to be allowable for at least the same reasons discussed above with respect to Claim 23.

#### Claims 16, 18-21

Amended independent Claim 16 recites, in its entirety:

16. A computer-readable storage medium having stored thereon a data structure, the data structure comprising:

a first data field containing data representing a global initialize function;

LAW OFFICES OF  
CHRISTENSEN O'CONNOR JOHNSON KINDNESS<sup>PLLC</sup>  
1420 Fifth Avenue  
Suite 2800  
Seattle, Washington 98101  
206.682.8100

a second data field containing data representing a global terminate function; and

a third data field containing data representing an application function.  
(Emphasis added.)

Claim 16 is submitted to be allowable for at least the same reasons discussed above with respect to Claim 24.

Claims 18-21 depend from Claim 16 and are submitted to be allowable for at least the same reasons discussed above with respect to Claim 16. Additionally, Claims 18-21 are submitted to be allowable for at least the same additional reasons discussed above with respect to Claims 25-28.

Claims 1, 3, 5-8, and 10

Claim 1 recites, in its entirety:

1. A method for using a framework module to run an application, the framework module comprising an application table and a parameter table, the application table comprising one or more application table entries, the parameter table comprising one or more parameter table entries, the method comprising:

selecting an application table entry; and

processing the selected application table entry, the processing comprising:

running a global initialize function referenced by the selected application table entry, running a sub-application referenced by the selected application table entry with one or more parameters referenced by one or more parameter table entries, and running a global terminate function referenced by the selected application table entry. (Emphasis added.)

In general, Claim 1 is submitted to be allowable for at least the same reasons discussed above with respect to Claims 16 and 23, since Kossatchev does not teach or suggest a framework module to run an application where the framework comprises an application table and a parameter table (Claim 23) or global functionality (Claim 16). Kossatchev discloses specifying the "behavior" of a group of parallel procedures to be tested in a parallel mode separately from

consecutive procedures." Kossatchev, Col. 1, lines 44-46. (Emphasis added.) Kossatchev does not teach or suggest running a global initialize function referenced by the selected application table entry. Kossatchev also does not teach or suggest running a sub-application referenced by the selected application table entry with parameters referenced by parameter table entries. Kossatchev discloses in Figure 2 a means 12 for generating specifications of the procedure interface 4 and a test source generator 14, which generates test source code based on the specifications. The specifications and test sources are stored in a repository 16. Kossatchev, Col. 3, lines 6-13. This is in contrast to Claim 1, which recites the use of an application table and a parameter table containing references to sub-applications and corresponding parameters, respectively, to run the sub-applications. Therefore, Claim 1 is further submitted to be allowable for the additional reasons discussed above.

Claim 3 depends from Claim 1 and is submitted to be allowable for at least the same reasons discussed above with respect to Claim 1.

Claims 5-8 depend from Claim 1 and are submitted to be allowable for at least the same reasons discussed above with respect to Claim 1.

Claim 10 depends from Claim 1 and is submitted to be allowable for at least the same reasons discussed above with respect to Claim 1.

#### Claims 11 and 14-15

Independent Claim 11 recites, in its entirety:

11. A method for building a framework module for running an application, the framework module comprising an application table and a parameter table, the application comprising one or more sub-applications, the method comprising:

collecting data specifying one or more sub-applications composing the application;

collecting data specifying one or more parameters to the one or more sub-applications;

creating the application table, the creating of the application table comprising creating an application table entry for each of the one or more specified sub-applications, the creating of an application table entry comprising creating a reference to a global initialize function, creating a reference to a global terminate function, and creating a reference to the sub-application; and

creating the parameter table, the creating of the parameter table comprising creating a parameter table entry for each of the one or more specified sub-application parameters, the creating of a parameter table entry comprising creating a reference to a name of the parameter and creating a reference to a type of the parameter. (Emphasis added.)

In general, Claim 11 is submitted to be allowable for at least the same reasons discussed above with respect to Claims 1, 16, and 23.

Claims 14 and 15 depend from Claim 11 and are submitted to be allowable for at least the same reasons discussed above with respect to Claim 11.

Rejection of Claims 2, 4, 9, 12, 13, 17, 29, and 34 under 35 U.S.C. § 103(a)

As noted above, Claims 2, 4, 9, 12, 13, 17, 29, and 34 were rejected under 35 U.S.C. § 103(a) by the Office Action for "as being unpatentable over Kossatchev...in view of Grey." Each set of claims as grouped by the Office Action on pages 8 and 9 are addressed separately as follows.

Claims 2, 12, and 17

Claim 2 depends from Claim 1 and is submitted to be allowable for at least the same reasons discussed above with respect to Claim 1. As noted above with respect to Claim 1, Kossatchev does not teach or suggest a global initialization function referenced by the selected application table entry. Grey fails to supply the teachings missing from Kossatchev. Grey discloses that "an unnamed synchronization object may be created that the user can access through an ActiveX reference variable." (Emphasis added.) Grey, Col. 23, lines 24-26. Grey

further discloses that the "user can use the ActiveX reference to the object in place of its name when performing operations on the Object...without performing a Create operation in each thread." Grey, Col. 23, lines 62-64. The ActiveX reference variable is used by Grey to access a synchronization object, in contrast to Claims 1 and 2, which recite the use of an application table entry to run a global initialize function. Additionally, Claim 2 recites "at least one of the global initialize and the global terminate functions is a NULL function." (Emphasis added.) In contrast, Grey discloses that "if the user specifies an empty string as the name for a synchronization object, then an unnamed synchronization object may be created that the user can access through an ActiveX reference variable." Grey, Col. 23, lines 23-26. Claim 2 recites a NULL function, in contrast to Grey, which discloses an empty string as the name for a synchronization object, which is different from a function. Therefore, Claim 2 is further submitted to be allowable for the additional reasons discussed above.

Claims 12 and 17 are submitted to be allowable for at least the same reasons discussed above with respect to Claim 2.

Claims 4, 13, 22, and 29

Claim 4 depends from Claim 1 and is submitted to be allowable for at least the same reasons discussed above with respect to Claim 1. As noted above with respect to Claim 2, Kossatchev does not teach or suggest a global initialization function referenced by the selected application table entry. Grey fails to supply the teachings missing from Kossatchev. Grey discloses that "an unnamed synchronization object may be created that the user can access through an ActiveX reference variable." (Emphasis added.) Grey, Col. 3, lines 24-26. Claim 4 recites, *inter alia*, "running a thread initialize function referenced by the selected application table entry." (Emphasis added.) Grey discloses a method that is used during the development of a program using an application development environment. During the development of the



application, the developer may specify a batch synchronization section. Grey, Col. 12, lines 21-29. This is in contrast to Claim 2, which recites running a function referenced by the selected application table entry in a run-time environment. Therefore, Claim 4 is further submitted to be allowable for the additional reasons discussed above.

Claim 13 depends from Claim 11 and is submitted to be allowable for at least the same reasons discussed above with respect to Claim 11. As noted above with respect to Claim 11, Kossatchev does not teach or suggest running a global initialize function referenced by the selected application table entry. Kossatchev also does not teach or suggest running a sub-application referenced by the selected application table entry with parameters referenced by parameter table entries. Grey fails to supply the teachings missing from Kossatchev. Grey discloses a method whereby each thread in the plurality of threads may execute the program until the thread arrives at the enter point for the batch synchronization section. The thread is blocked upon arriving at the enter point until all other threads arrive at the enter point. Once all threads have arrived at the enter point, execution of the program within the batch synchronization section proceeds. Grey, Col. 5, lines 52-58. Grey discloses a method for creating a computer program using batch synchronization section, whereas Claim 13 recites creating a reference to a thread initialize function and thread terminate function. Therefore, Claim 13 is further submitted to be allowable for the additional reasons discussed above.

Claim 22 recites features similar to Claim 13 and is, therefore, submitted to be allowable for at least the same reasons discussed above with respect to Claim 13.

Claim 29 recites features similar to Claim 13 and is, therefore, submitted to be allowable for at least the same reasons discussed above with respect to Claim 13.

Claim 9 depends from Claim 1 through Claim 8 and is submitted to be allowable for at least the same reasons discussed above with respect to Claim 2, which also depends from Claim 1.

Claim 34 depends from Claim 23 and is submitted to be allowable for at least the same reasons discussed above with respect to Claim 29, which also depends from Claim 23.

CONCLUSION

In summary, applicants respectfully submit that all the claims in this application are clearly allowable in view of the disclosures of Kossatchev and Grey, applied singly or in any motivated combination. As a result, applicants respectfully request that all of the claims remaining in this application be allowed and this application be passed to issue. If the Examiner has any questions, the Examiner is invited to contact applicants' attorney at the number set forth below.

Respectfully submitted,

CHRISTENSEN O'CONNOR  
JOHNSON KINDNESS<sup>PLLC</sup>

  
Gary S. Kindness  
Registration No. 22,178  
Direct Dial No. 206.695.1702

I hereby certify that this correspondence is being deposited with the U.S. Postal Service in a sealed envelope as first class mail with postage thereon fully prepaid and addressed to Mail Stop AF, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the below date.

Date: March 30, 2006 Rene W. Wedon - Treasurer

GSK/FXM:pww/aew

LAW OFFICES OF  
CHRISTENSEN O'CONNOR JOHNSON KINDNESS<sup>PLLC</sup>  
1420 Fifth Avenue  
Suite 2800  
Seattle, Washington 98101  
206.682.8100